2. **High-Performance Superscalar-Based Computer System with Out-of-Order Instruction Execution and Concurrent Results Distribution**, Appl. No. 08/397,016, filed March 1, 1995, now U.S. Patent No. 5,560,032, by Quang Trang *et al.*;

3. **RISC Microprocessor Architecture with Isolated Architectural Dependencies**, Appl. No. 08/292,177, filed August 18, 1994, now abandoned, which is an FWC of Appl. No. 07/817,807, filed January 8, 1992, which is a continuation of Appl. No. 07/726,744, filed July 8, 1991, by Yoshiyuki Miyayama;

4. **RISC Microprocessor Architecture Implementing Fast Trap and Exception State**, Appl. No. 08/345,333, filed November 21, 1994, now U.S. Patent No. 5,481,685, by Quang Trang;

5. **Page Printer Controller Including a Single Chip Superscalar Microprocessor with Graphics Functional Units**, Appl. No. 08/267,646, filed June 28, 1994, now U.S. Patent No. 5,394,515, by Derek Lentz *et al.*; and

6. **Microprocessor Architecture Capable with a Switch Network for Data Transfer Between Cache, Memory Port, and IOU**, Appl. No. 07/726,893, filed July 8, 1991, now U.S. Patent No. 5,440,752, by Derek Lentz *et al.*.--

Page 2, please delete lines 1-11.

In the Claims:

Please add the following new claims 31 to 45.

-31. In a data processing system, which includes a central processing unit (CPU) that performs operations by executing instructions, a data register system comprising:

a first register set including a plurality of first registers each for holding integer data;

a second register set including a plurality of second registers each for holding integer data or floating point data, wherein a specific instruction includes a field specifying which of said first and second register sets is to be accessed in response to execution of said specific instruction; and

means, responsive to the field, for accessing said first register set or said second register set as specified by said field, including

i) reading means for reading an operand value from either the first register set or second register set as specified by said field, and

ii) writing mean for writing a result value to sad first register set or said second register set as specified by said field.

32. The apparatus of claim 31, wherein said first and second register sets each have two write ports and five read ports.

33. The apparatus of claim 31, further comprising execution means for executing said specific instruction, wherein said specific instruction performs an operation upon operands to generate a result, said specific instruction specifying a respective source address for each operand and a destination address for the result of said specific instruction.

34. The apparatus of claim 31, wherein said specific instruction can specify a first and a second source address and a destination address, with each address specifying either of the first or second register sets such that said specific instruction requires access to both register sets.

35. The apparatus of claim 34, wherein said means for accessing provides said specific instruction parallel access to both the first and second register sets.

36. An apparatus, for use with a data processing system that performs read operations and write operations upon data values of a first data type and a first data width, wherein the first data type is floating point, and upon data values of a second data type and a second data width different from the first data width, the second data type is integer, the data processing system specifying a read address and data type for each read and a write address and data content for each write, the apparatus comprising:

a register set including a plurality of individually addressable registers, each register being wide enough to hold a value of the first data type or the second data type;

read access means, responsive to the data processing system performing a given read operation of a specific data type, for accessing said register set to retrieve data from a given register, which is individually addressed at a specified read address of said given read operation,

write access means, responsive to the data processing system performing a given write operation, for accessing said register set to store into a given register, which is individually addressed at the specified write address of said given write operation, data specified by said write operation; and

wherein said read and write access means, respectively, retrieve and store data having the first data width responsive to the data processing system performing floating point operations, and data having the second data width responsive to the data processing system performing integer operations.

37. The apparatus of claim 36, wherein the first data width is sixty-four bits and the second data width is thirty-two bits.

38. The apparatus of claim 36, further comprising processing means for executing instructions including Boolean execution unit to execute Boolean combinational instructions each operating on one or more Boolean operands to generate a Boolean result, each Boolean combinational instruction including one or more Boolean fields specifying a location of each operand and result, integer execution unit. to execute integer instructions each operating on one or more integer operands to generate an integer result, each integer instruction including one or more integer fields specifying a location of each operand and result, and floating point execution unit to execute floating point instructions each operating one or more floating point operands to generate a floating point result, each floating point instruction including one or more floating point fields specifying a location of each operand and result.

39. The apparatus of claim 38, further comprising a Boolean register set having a plurality of Boolean registers, each Boolean register for holding one of said Boolean operands or Boolean results.

10 9

~~40.~~ The apparatus of claim ~~39~~, wherein said plurality of Boolean registers include:

(i) a first set of Boolean registers, and

(ii) a second set of Boolean registers;

means, coupled to said plurality of Boolean registers, for selecting said first or said second set of Boolean registers as a currently active set,

wherein said Boolean execution unit is responsive to said means for selecting and stores results into only said currently active set of said Boolean registers; and

means, responsive to execution of a given Boolean instruction by said Boolean execution unit, for storing the result of said given Boolean instruction into one of said Boolean registers, said one Boolean register being indicated by said given Boolean instruction as the destination of its Boolean result.

11 9

~~41.~~ The apparatus of claim ~~38~~, wherein said Boolean execution unit comprises:

numerical execution means for executing numerical comparison instructions to compare two multi-bit numerical operands and to accordingly produce a single-bit Boolean value result.

12

~~42.~~ An apparatus comprising:

integer execution means for executing integer instructions, each integer instruction performing an integer operation upon one or more integer value operands and generating an integer value result;

floating point execution means for executing floating point instructions, each floating point operation performing a floating point operation upon one or more floating point value operands and generating a floating point value result;

boolean execution means for executing boolean instructions, each boolean operation performing a boolean operation upon one or more boolean value operands and generating a boolean value result;

wherein each instruction specifies one or more sources from which its one or more operands are to be retrieved and further specifies a destination to which its result is to be stored, each operation also optionally specifying an integer value base and an integer value index;

a register bank including,

i) first register set, having a plurality of first registers, for holding integer values and floating point values;

ii) second register set, having a plurality of second registers, for holding integer values; and

iii) third register set, having a plurality of third registers, for holding Boolean values;

access means, coupled to said first register set, said second register set, said third register set and to all three execution means, for,

i) retrieving, from any one first register, an integer value operand for the integer execution means, a floating point value operand for the floating point execution means, or an integer value base or index for either execution means, as indicated by an instruction;

ii) storing, into any one first register, an integer value result from the integer execution means or a floating point value result from the floating point execution means, as indicated by an instruction;

iii) retrieving, from any one second register, an integer value operand for said integer execution means, or an integer value base or index for either execution means, as indicated by an instruction;

iv) storing, into any one second register, an integer value result from said integer execution means, as indicated by an instruction;

v) retrieving, from any one third register, a Boolean value operand for the Boolean execution means, as indicated by a Boolean combinational instruction, and

vi) storing, into any one third register, a Boolean value result from the Boolean execution means, as indicated by a Boolean combinational instruction.

13

~~43~~. An instruction execution unit of a RISC processor, comprising:

an execution engine that includes a data dependency checker, an integer functional unit, a floating point functional unit, a boolean functional unit, means for performing a context switch, and a mode control unit that provides mode information;

a register file, connected to said execution engine, having two or more register banks, each bank having,

48

an integer register set that includes, a first integer register set that includes a first subset of registers and a second subset of registers, a shadow integer register set and mode switch, wherein a first register in said first integer register set is set to zero, wherein said mode switch is switchingly coupled to said mode control unit, said first integer register set and said shadow integer register set, wherein said mode switch provides access to said second subset of registers or said shadow integer register set depending upon said mode information, whereby all access to said integer register set is via said mode switch,

a re-typable register set, wherein said re-typable register set can store floating point data or integer data, said re-typable register set including means for preventing accidental access of a floating point value as though it is an integer value or accidental access of a integer value as though it is a floating point value, wherein a first register in said re-typable register set is set to zero, and

a boolean register set that includes a condition status register (CSR) that includes a plurality of boolean registers, a previous condition status register (PCSR) that includes a plurality of boolean registers and a selector unit that is responsive to said mode information to select between said CSR and said PCSR, wherein a first boolean register in said CSR is set to zero,

wherein said data dependency checker allows a slave instruction to execute without delay when the result of a master instruction is said first register in said first integer register set.

14 ~~44.~~ The instruction execution unit of claim 13 ~~43~~, wherein said two or more register banks comprise the same hardware configuration .

15 ~~45.~~ The instruction execution unit of claim 13 ~~43~~, wherein said mode control unit is a processor status register that includes a flag to indicate whether interrupts are enable or disabled.

16 ~~46.~~ The instruction execution unit of claim 13 ~~43~~, wherein said CSR is used when interrupts are enabled and said PCSR is used when interrupts are disabled.

17 ~~47.~~ The instruction execution unit of claim 13 ~~43~~, wherein said PCSR is available to said execution engine as a special register.

49

18 13

~~48.~~ The instruction execution unit of claim ~~43~~, wherein said selector unit writes the contents of said CSR into said PCSR when interrupts are disabled, overwriting the former contents of said PCSR, and when interrupts are re-enabled, said selector unit writes the contents of said PCSR into said CSR.

19 13

~~49.~~ The instruction execution unit of claim ~~43~~, wherein said boolean register set does not include dedicated condition flags.

20 13

~~50.~~ The instruction execution unit of claim ~~43~~, wherein said execution engine operates on one or more instructions having a destination field that specifies said integer register set or said re-typable register set, an opcode, a first operand field that specifies said integer register set or said re-typable register set, and a second operand field that specifies said integer register set or said re-typable register set.

21 13

~~51.~~ The instruction execution unit of claim ~~43~~, wherein said integer functional unit and said floating point functional unit are configured to execute a comparison instruction, wherein a destination of said comparison instruction is said boolean register set.

22 13

~~52.~~ The instruction execution unit of claim ~~43~~, wherein said instruction further includes a field indicating an addressing mode.

23 13

~~53.~~ The instruction execution unit of claim ~~43~~, wherein said floating point functional unit is configured to perform floating point and integer operations.

24 13

~~54.~~ The instruction execution unit of claim ~~43~~, wherein said integer functional unit, said floating point functional unit and said boolean functional unit are each configured to execute a plurality of instructions simultaneously.

25 13

~~55.~~ The instruction execution unit of claim ~~43~~, further comprising a switching and multiplexing control unit connected between said integer functional unit and said register file, wherein said integer functional unit executes instructions having at least one source and at least

50

one destination, each instruction indicating whether said at least one source and said at least one destination reside in said integer register set or said re-typable register set.

26 25
~~56~~. The instruction execution unit of claim ~~55~~, wherein said mode switch connects said switching and multiplexing control unit to said shadow integer register set when interrupts are disabled.

27 26
~~57~~. The instruction execution unit of claim ~~56~~, wherein said mode control unit is a processor status register that includes a flag that indicates whether interrupts are ~~enable~~ enabled or disabled.

28 13
~~58~~. The instruction execution unit of claim ~~43~~, further comprising a switching and multiplexing control unit connected between said floating point functional unit and said register file, wherein said floating point functional unit executes instructions having at least one source and at least one destination, each instruction indicating whether said at least one source and said at least one destination reside in said integer register set or said re-typable register set.

29 13
~~59~~. The instruction execution unit of claim ~~43~~, wherein said execution engine executes an instruction having at least one source and at least one destination, wherein said instruction further includes a field that indicates which of said two or more banks has stored therein said one or more source and where said destination is located within said two or more banks.

30 13
~~60~~. The instruction execution unit of claim ~~43~~, further includes a bank selector unit that selects between said two or more register banks.--

---

Please cancel claim 1 without prejudice or disclaimer.